WHITEBOARD
PEOPLE

Test Driving an Organisation

Paul O'Keeffe

Shaun Loh

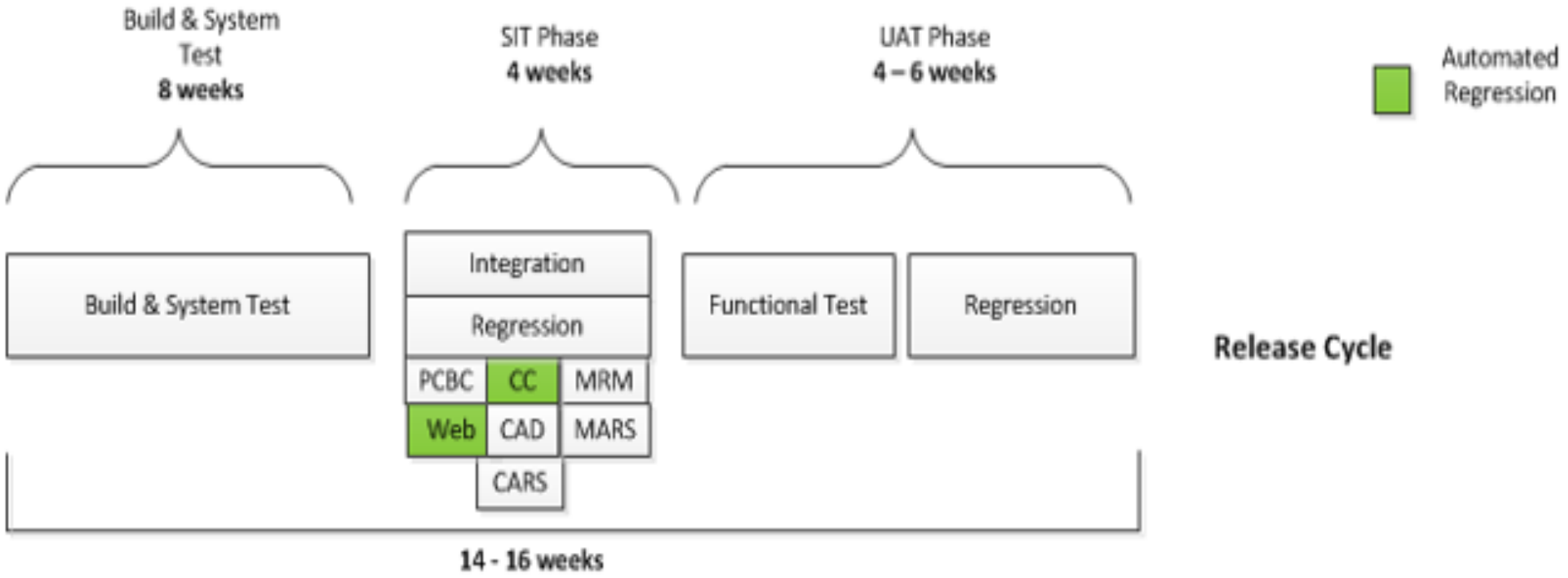
We Have a Problem

- Core applications
- Business ideas slow to get to market
- Manual testing times dominated the release cycle
- Long tail test cycles delaying release
- Defect detection late in the process
- Some defects getting to production
- Multiple independent test teams sharing little
- Lack of trust in tests due to inconsistent coverage and execution

Quality / Speed Delivery

- Quality initiative
- Morphed into “Speed Delivery”
- Cycle time is perhaps the most important metric in software development
- Reduce the time between change request and production delivery

Current State



GUI Test Automation

- Web applications
- Desktop application
- Tight coupling between applications
- Initially build a safety net
- Then build out application coverage
- Business logic in the GUIs was rife so we couldn’t sensibly test at the API level or even at the HTTP level

WHITEBOARD
PEOPLE

What Test Pyramid?

- The test pyramid minus the base and middle;
  - and the top
- We build the top, because the base and middle couldn't be
- Code was not designed to be testable
  - Frameworks!
- Swiss cheese pyramid
- Normally, building a few happy days scenarios at the top of the pyramid is the right approach, but without a good base or middle, fragility is worse
- So many things would break tests
  - Who / what to blame?

What Could Possibly Go Wrong?

- Infrastructure
- Environment
- Data
- Applications
- Everything…
- Stabilising a GUI test automation suite is incredibly hard work

Infrastructure

- Cloud (AWS)
- But production still on premise, so different
- Long lived snowflakes
- Provisioning was gradually being automated
- Both for testing and application infrastructure

WHITEBOARD
PEOPLE

Controllable Environment

- Nail down environment on build agents precisely
- Spread out into nailing down everything external which is changing and/or could cause problems
  - Hardware specs
  - Manual testers sharing environments
  - Batch jobs
  - Software updates
- Limited due to ownership by other teams
  - Separation of acceptance testing from development teams

WHITEBOARD PEOPLE

Data

- Don’t rely on a copy of production data
  - Because you can’t rely on it
- Always generate the data you need for a test
- Production data and batch jobs taking too long to run in under specified environments

WHITEBOARD
PEOPLE

Test Isolation

- Creating a new member for each test
- Natural isolation
  - Data was completely scoped by member
- Ensures each test is completely isolated from any other

WHITEBOARD
PEOPLE

Applications (with Real Users)

WHITEBOARD PEOPLE

- Users are apologists for poor application behavior
  - Robots aren't
- Real users seemed to have been trained to have a high tolerance for slow application responses
- Synchronous business processes built on top of asynchronous systems
- Users just refresh until things have finished
- Real users worked around application errors like the every 10th index problem
- Doesn't make test automation any easier, though

Application Driver Layer

- Application models
- Page models
- Grow this as you need it for the tests you've written
- Don't put a hard cap on it too soon
  - Cucumber/Gherkin
  - Commercial tools
  - Other open source tools (who shall remain nameless)
- Domain, not UI

WHITEBOARD
PEOPLE

Application Driver Layer

- Most tools put a hard cap on application modelling, usually at the page level
- Need more flexibility. Not everything is shaped like a cucumber
- Renewals model is a good example
- Code retains the flexibility, since it has the necessary mechanisms for abstraction
- Ask yourself whether your tool can compete with code on this front?

Page Models

- Disappointing lack of free tools to do basic page modelling as code
- Learned a lot about how to do good page modelling in Java and C#
- Some advances on the currently accepted state of the art IMHO
- Design of page model DSL prevented run-time errors and enabled auto completion
- Project coming soon with some of our designs
- DSLs for APIs and GUIs. Sometimes the same one so we could switch the two easily

WHITEBOARD
PEOPLE

Asynchronous Behaviour

- No fixed delays (sleeps!)
- Polling loops
- Events

WHITEBOARD
PEOPLE

Weapon of Mass Destruction

Effect on Systems Under Test

- Weapon of mass destruction
- Unexpected loads / greater scale
  - Running our tests caused a higher load than production
  - Due to having 30 build agents all running tests faster than normal users
- But test environments are much scaled down versions of production to cut costs
- Single server rather than clusters (multicast problem)
- Data sizes?
  - Insignificant compared to production
  - But growth over time without reset caused issues

Impact

- Manual testing is usually pretty low impact
- Automated testing is high impact
- Continuous Integration is even more impact
- Existing systems may only be designed to handle manual tests
- Testing performance/scale demands may be higher than production!!!
  - Especially if you want fast development cycles
- Force rest of the organisation to change / improve

WHITEBOARD
PEOPLE

Test Execution

- Need a tool with the right granularity for GUI test automation.
- Run each test and collect results would be about right.
- CI server to coarse. Run all tests in one build.
- Test grid too fine. Run each UI action remotely from a central controller.
- Broke things up by having a build per test on CI server, but not well supported.

WHITEBOARD
PEOPLE

Continuous Integration Server

- As opposed to the practice of continuous integration
- Support for long running acceptance tests
- Scalability of configuration
- New generation: pipeline as code (or at least better configuration)
  - GUIs hamper this

WHITEBOARD
PEOPLE

MSHTA

- The Uncanny Valley of web applications
- MSHTA applications are a pain to test
- Nearly a web app but not quite so you need special tools
- Framework didn’t allow application to be run in a browser just for testing
- Would be good to have a free tool which could do this as well as Selenium does for web
- MSHTA have some differences
  - Often more dialog window wizards
  - Locked down browser
  - Fewer controls
- Test automation forces organisation away from this style of application

WHITEBOARD
PEOPLE

Recording Evidence

- Video recording >
- Sequence of screen shots >
- Final screen shot >
- Logs only >
- Nothing
- Some problems are hard to reproduce just by running the test again
- Observability
- Convincing evidence for others within the organization
- Other benefits

Licensing

- Per machine licensing schemes are painful for desktop test development, but more so for a grid of build agents in the cloud
- Do not buy tools with licences which discourage the number of potential environments
- Site licences are fine

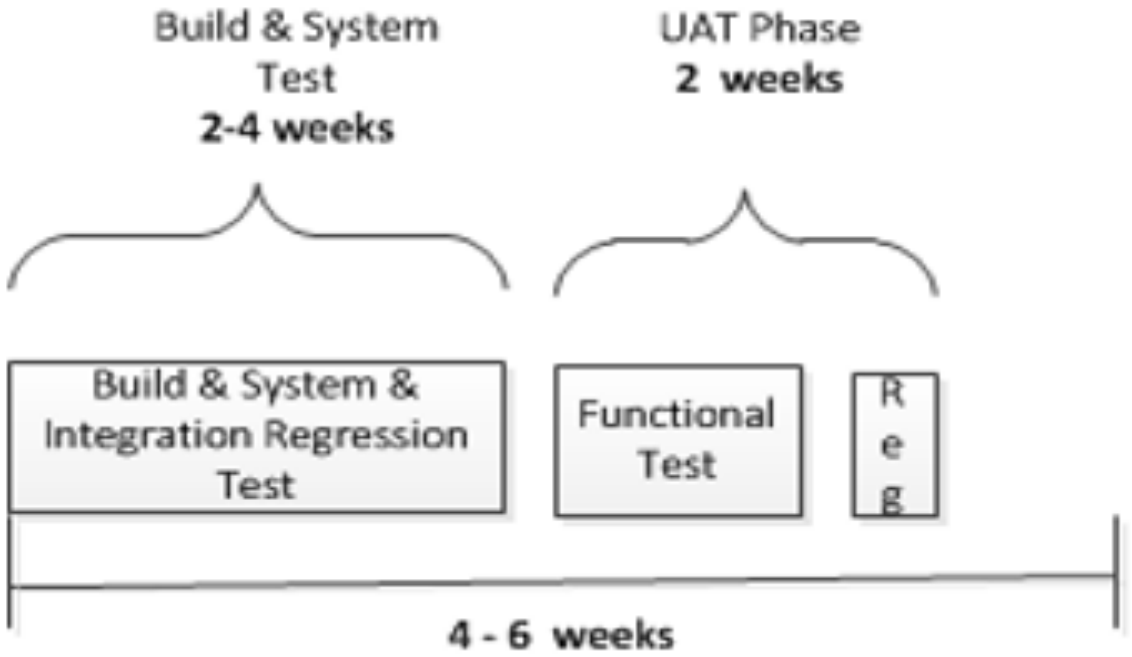
WHITEBOARD
PEOPLE

Results

- Shortened release cycles by many weeks
- Increased confidence to perform releases, where they might not otherwise have been attempted
- But other changes are slow

WHITEBOARD
PEOPLE

New State


Build & System
Test
2-4 weeks
UAT Phase
2 weeks
Build & System &
Integration Regression
Test
Functional
Test
R
e
g
4 - 6 weeks

Test Driving the Organisation

WHITEBOARD PEOPLE

- TDD a whole organisation
  - Conway's Law, sort of
  - Can make moves and reorganise more easily with the safety net
- Removal of redundant delivery/testing phases
- Small environments
- Faster cycle times
  - Each change makes this almost more possible
  - Lower overhead
  - Cost saving
  - Earlier testing
  - Shift Left

Opportunity: Data Load

- Built tools for manual testers to set up data for exploratory testing
- Also for other manual tests we hadn't yet automated
- Removed dependence on production data and finding appropriate examples
- Where possible, used APIs for data load in our test setup
- Chicken and egg
  - No longer any need for production data, so we changed the organisation in a way which solved our own problem and several others
- Learning and Development (training data)
- Pushed development teams to open up APIs

Automate (Almost) Everything

- Analysis tools like Dashy and Helpy.
- Don’t just automate your tests, automate your analysis and anything else repetitive you have to do.

WHITEBOARD
PEOPLE

Test Execution

- Run your tests all the time?
  - Tradeoffs
- At least when anything changes, which in our case was pretty much all the time

WHITEBOARD
PEOPLE

Deployment and Other Automation

- Started pushing things towards automated deployments
- We would help out with this, since we had automation skills
- Software development skills of team flowed over into small product development
- Robot Process Automation (RPA)

Think Different

- Developer view of test automation
- Fresh eyes
- Embrace code
- Improved quality of test code
- Bring testing and development closer together

WHITEBOARD
PEOPLE

Way of Working

- Agile, but only within our small part of the SDLC
- Collaborative
- Pairing on demand
  - Moar pairing!!!
  - Particularly where increase skills or share knowledge, where they are unevenly distributed within the team
- Honest retrospectives
- Continuous improvement actions actually acted upon
- Attempted to influence other teams
- As far as we could go within the constraints of the organization

WHITEBOARD
PEOPLE

Invest in your People

- Super-interns
- And others with complementary skills from elsewhere in the organisation
- Limited budget
- Negotiated to get really good interns and then train them well
- People don't need to be an exact match for required skills
- Some programming ability is welcome, but no need for lots
- Can come from other IT background
  - Good to have alternative perspectives (e.g. support)
- Need to be keen to learn
- Can get fantastic leverage at low cost. We did.

Invest in your People

- Axes of learning/growth
  - Mature software development
  - Automated testing
  - Enterprise training/constraints
- Highly collaborative environment/team
  - Work with others with development/testing/domain/application experience
  - Balance of perhaps 50% of team already experienced

WHITEBOARD
PEOPLE

Invest in Your People

- Freedom to fail and learn
  - Psychological safety
  - Gives people confidence to try new things earlier
- Hands on immediately
- Interns became the mentors
- Grow existing people from the organisation
  - Graft them onto a new plant to reignite their motivation and grow new skills
- Chocolate